Significant Event System Tutorial: DAQ Operations

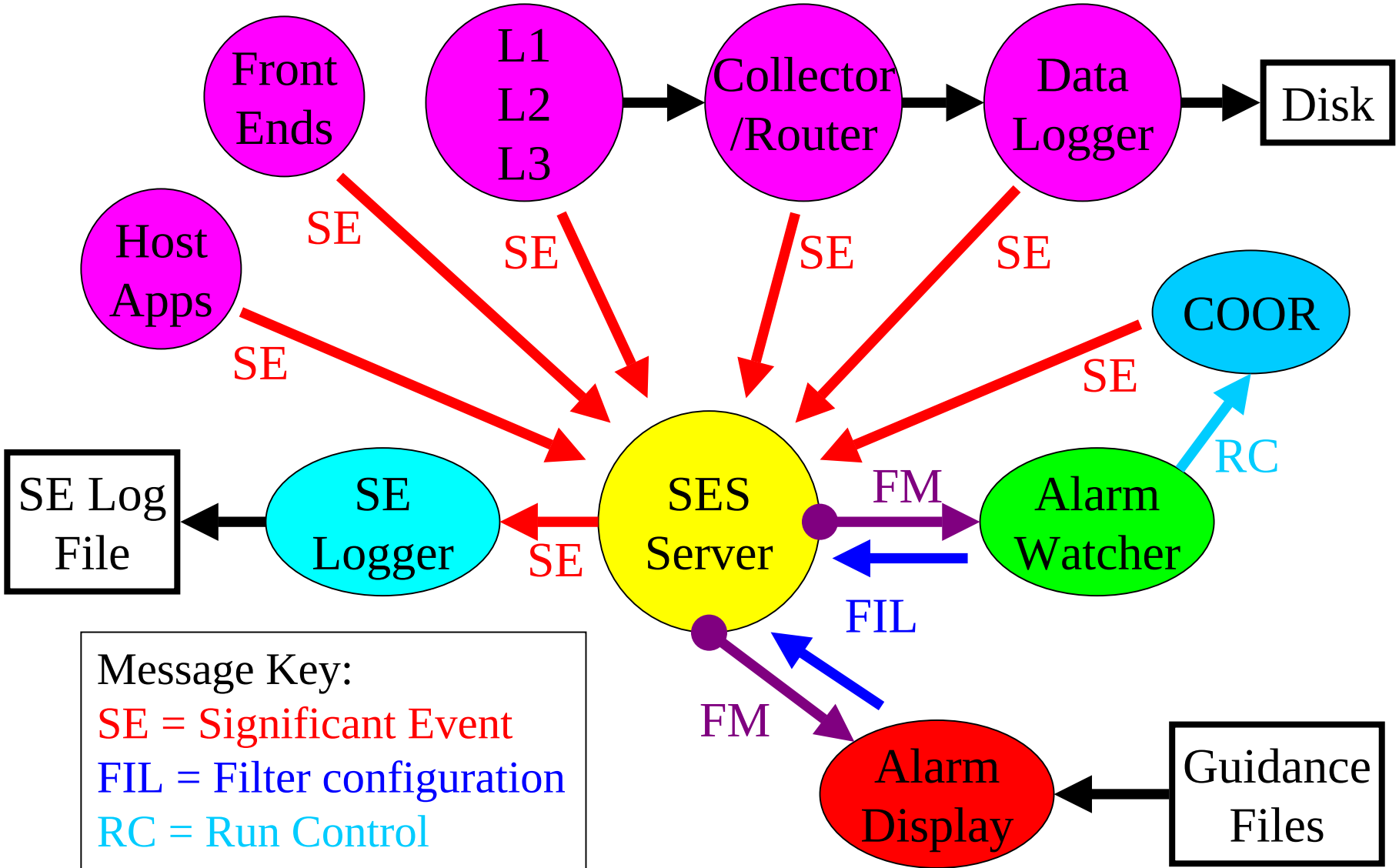
Geoff Savage
DZero Controls Group
May 2004

SES Tutorial Outline

- Purpose – the role of the SES
- Overview – SES system architecture
- Shift operations – alarm display
- Core applications – server, alarm watcher, logger
- Operation – starting, stopping, and status

Goals

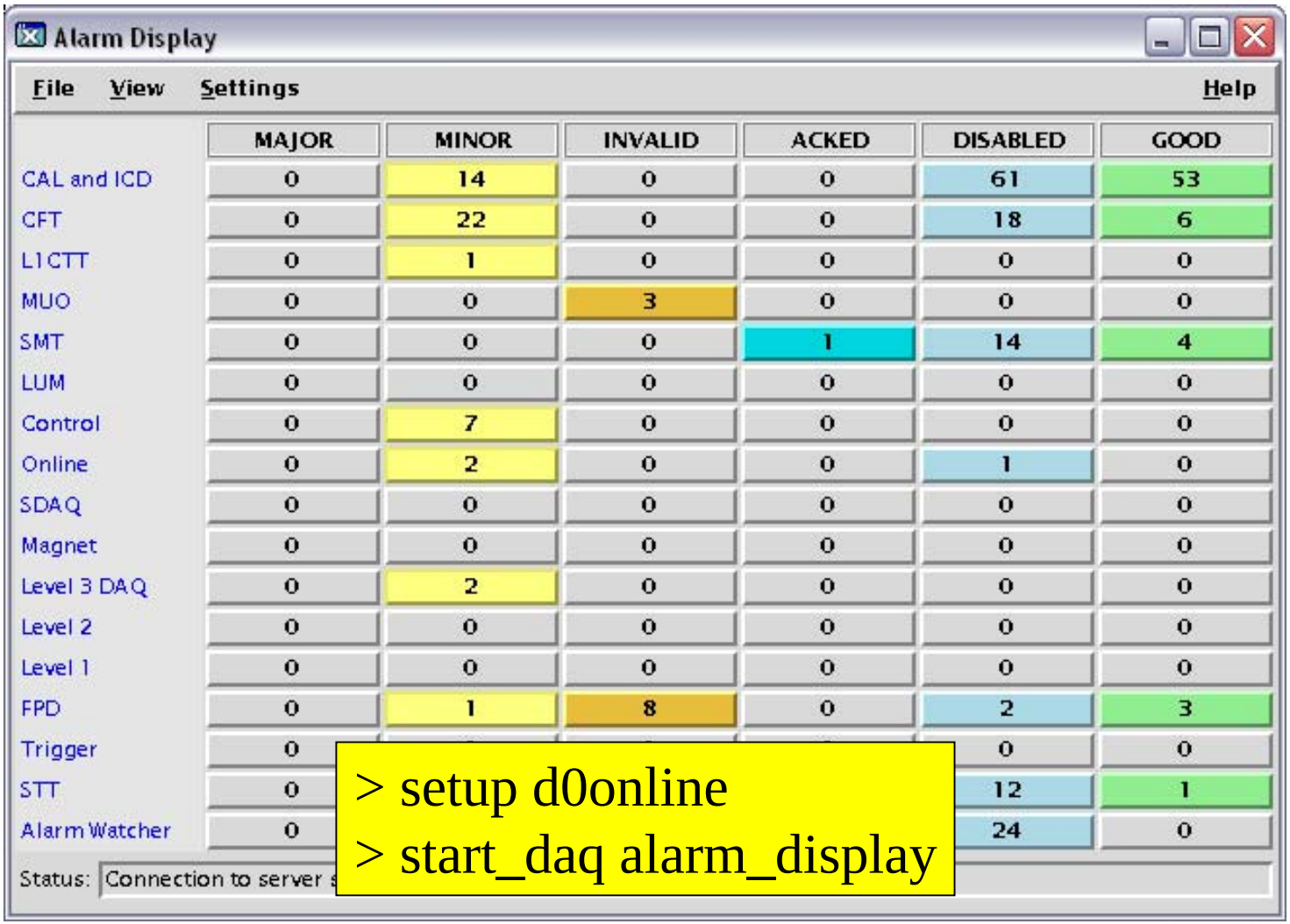
- Monitor the health of the DZero online system
- During detector operations produce, distribute, and display events which are significant to the experiment
  - Alarm conditions
  - DAQ state transitions from COOR
- Archive SE messages for later review
  - Look for trends when diagnosing equipment issues
  - In run I the detectors state was checked for all top event candidates to insure the events were not artifacts of a detector problem

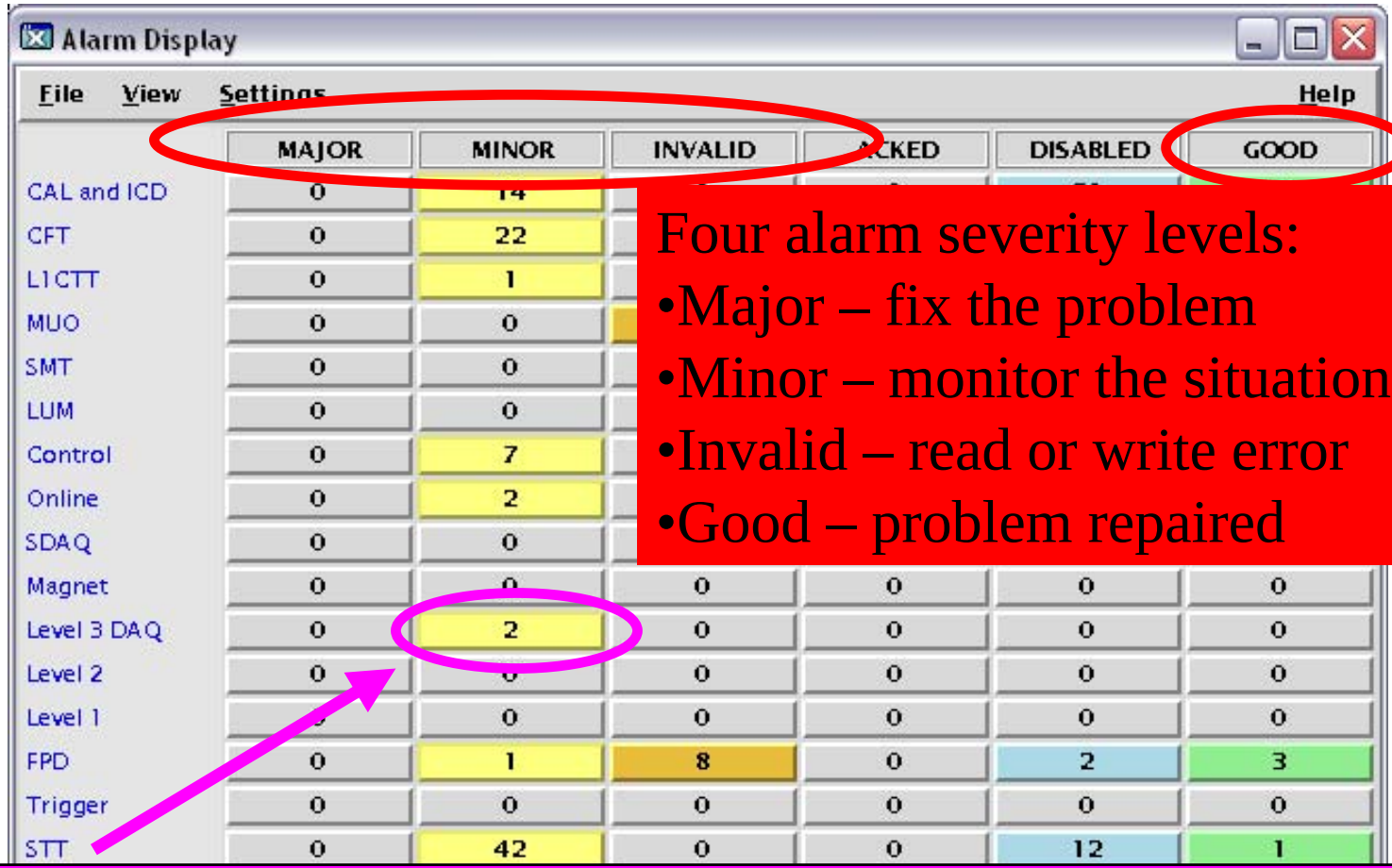

Front Ends
L1 L2 L3
Collector /Router
Data Logger
Disk
Host Apps
SE
SE
SE
SE
SE
SE
COOR
RC
SE Log File
SE Logger
SE
SES Server
FM
Alarm Watcher
FIL
FM
Alarm Display
Guidance Files
Message Key:
SE = Significant Event
FIL = Filter configuration
RC = Run Control
FM = Filtered SE Message
● = Filter

Functionality

- Sender Clients
  - Identify bad states
  - Send alarms
- Server
  - Maintains the current state
  - Supplies the state to clients on request
- Logger
  - Writes all SE messages to files
- Alarm watcher
  - Detects alarms that should pause runs
  - Sends run pause commands to COOR
- Alarm display
  - Shows the current state to the user
  - Access guidance to assist in resolving an alarm condition

Alarm Display



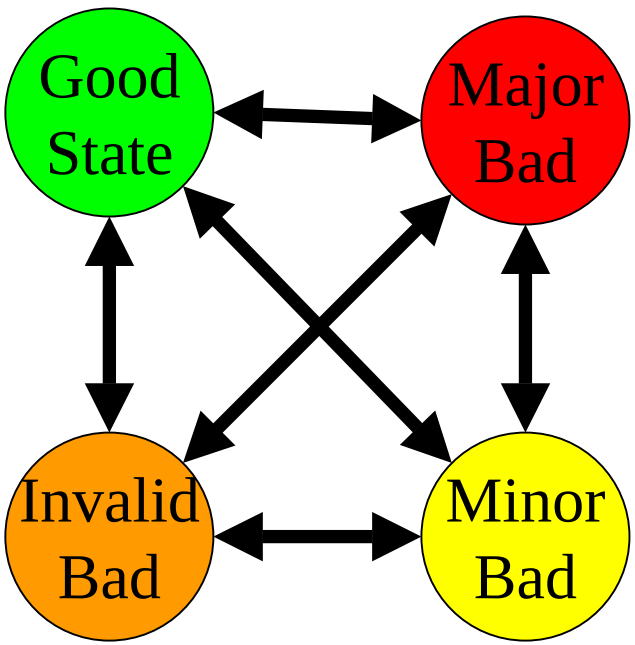


- Each button is labeled with a value that reflects the number of alarms of a severity that pass the filter for that row
- Each row has a different filter
- Alarms that pass multiple filters appear in multiple rows

Alarm Display

File View Settings Help

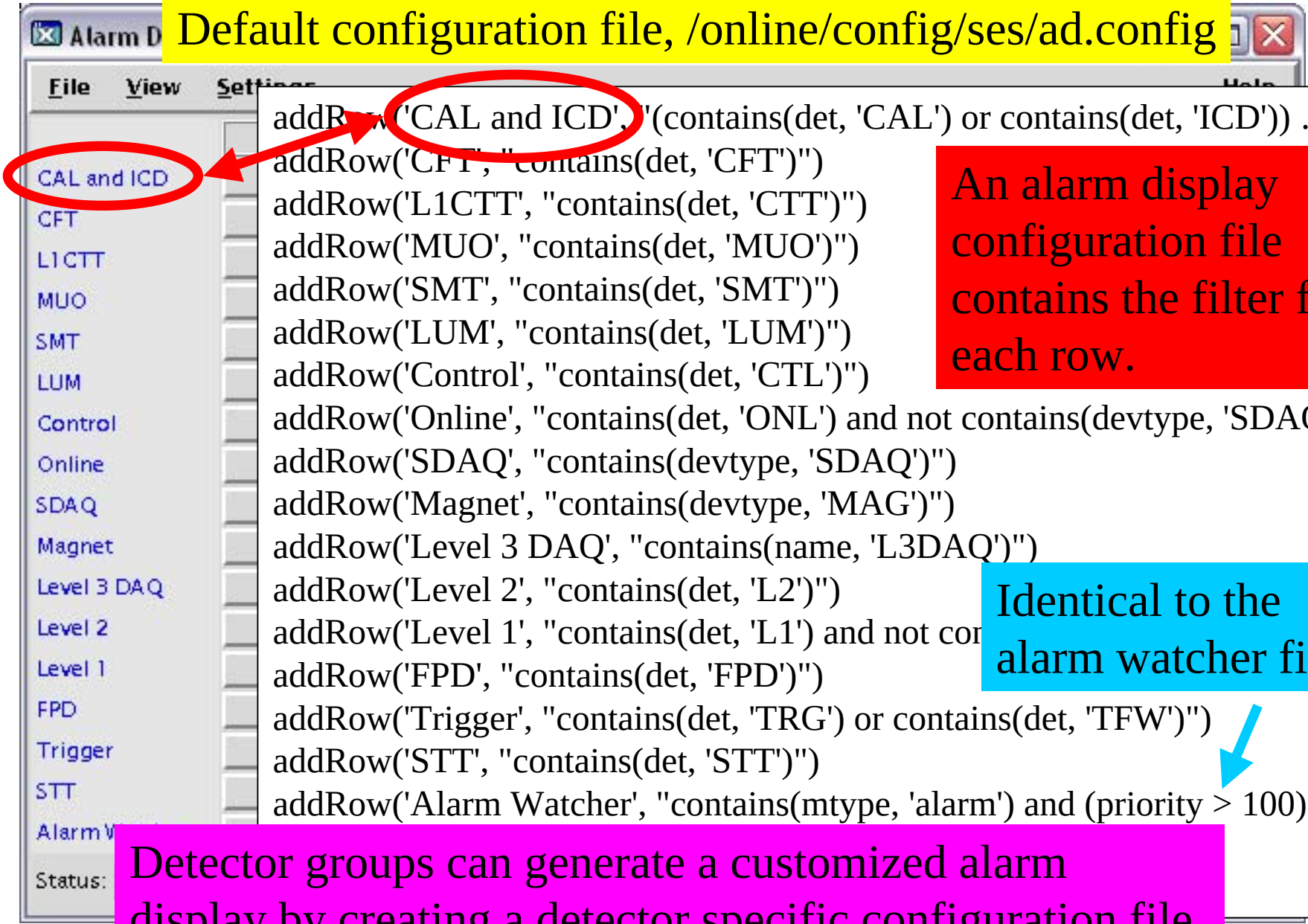
	MAJOR	MINOR	INVALID	ACKED	DISABLED	GOOD
CAL and ICD	0	14	0	0	61	53
CFT	0	22	0	0	18	6
L1 CTT	0	1	0	0	0	0
MUO	0	0	3	0	0	0
SMT	0	0	0	1	14	4
LUM	0	0	0	0	0	0
Control	0	7	0	0	0	0



Alarm severity transitions

- A device starts in the good state (not listed on the display)
- A bad alarm is in one of the three severity levels
- Transitions from the bad severities to good can occur at any point
- As can transitions between severities

Default configuration file, /online/config/ses/ad.config

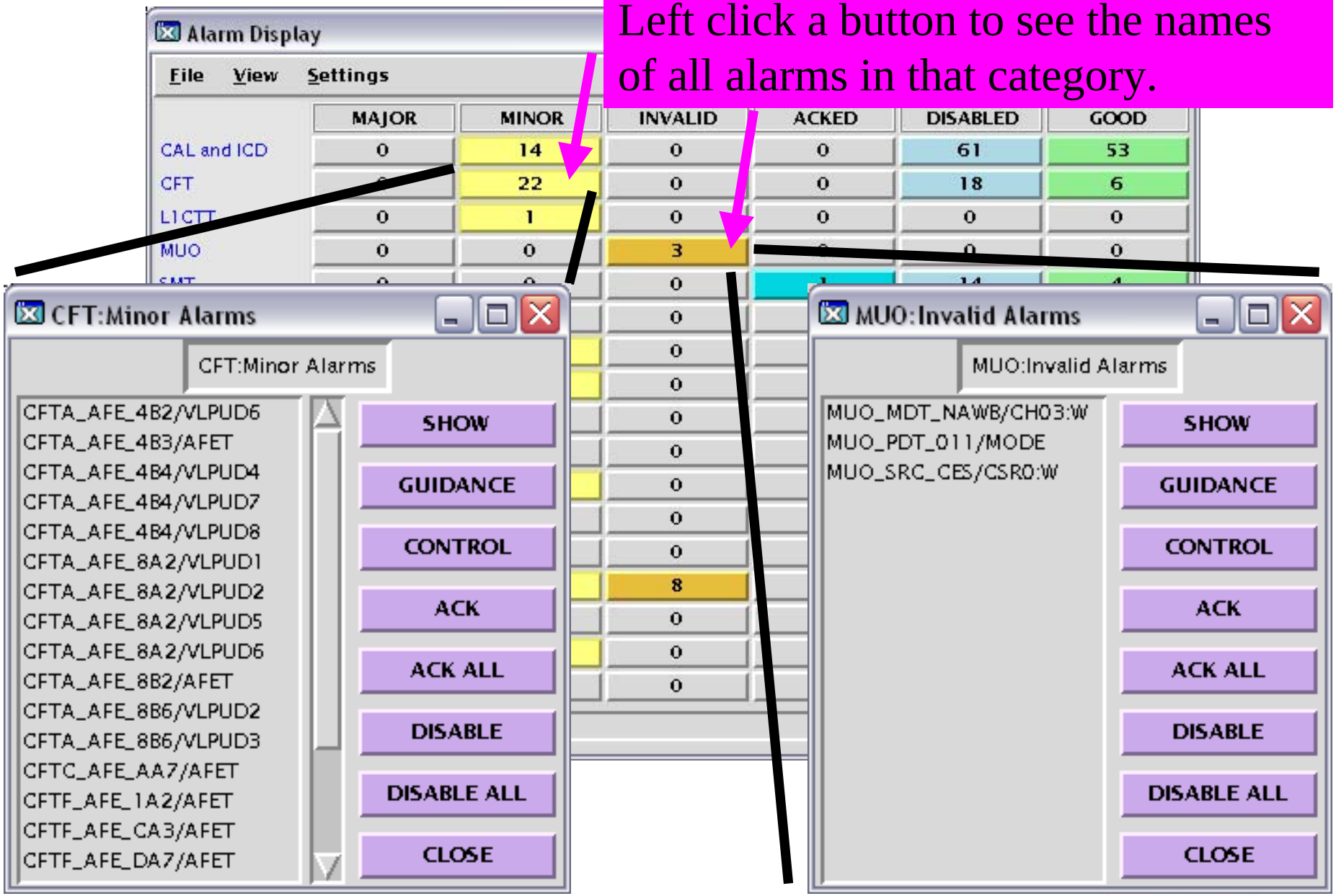


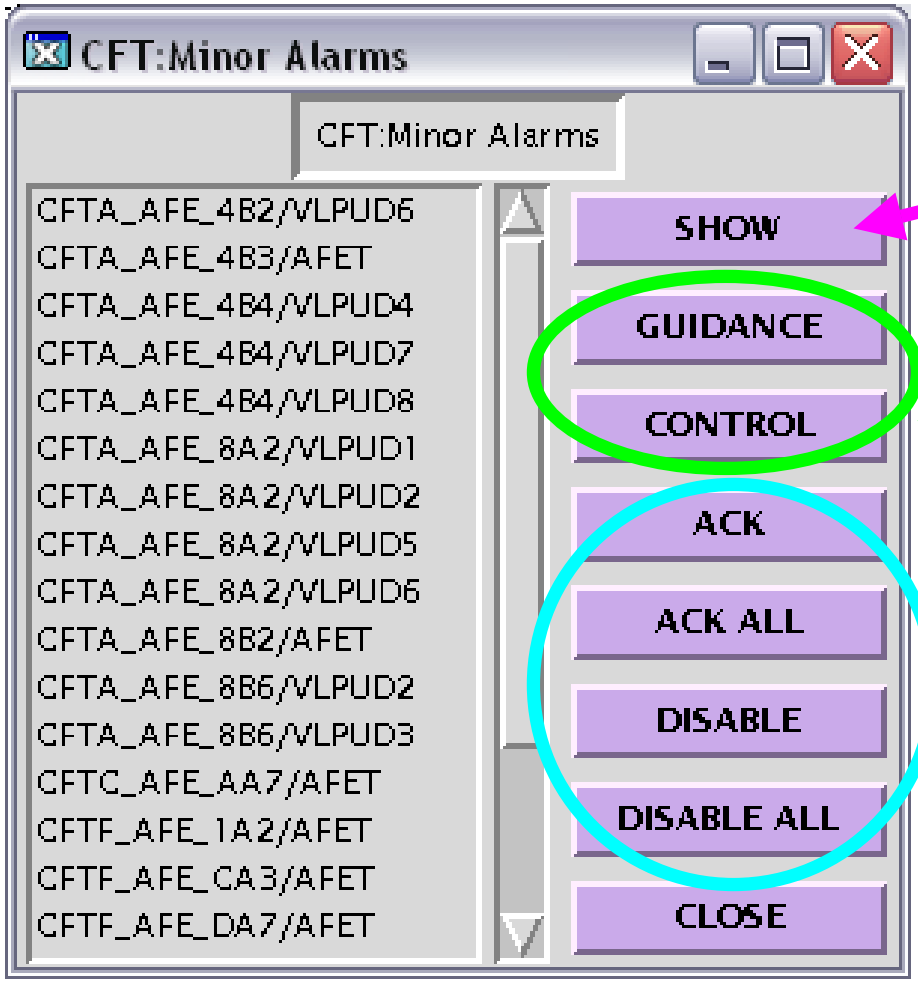

```
addRow('CAL and ICD', "(contains(det, 'CAL') or contains(det, 'ICD')) …")
addRow('CFT', "contains(det, 'CFT')")
addRow('L1CTT', "contains(det, 'CTT')")
addRow('MUO', "contains(det, 'MUO')")
addRow('SMT', "contains(det, 'SMT')")
addRow('LUM', "contains(det, 'LUM')")
addRow('Control', "contains(det, 'CTL')")
addRow('Online', "contains(det, 'ONL') and not contains(devtype, 'SDAQ')")
addRow('SDAQ', "contains(devtype, 'SDAQ')")
addRow('Magnet', "contains(devtype, 'MAG')")
addRow('Level 3 DAQ', "contains(name, 'L3DAQ')")
addRow('Level 2', "contains(det, 'L2')")
addRow('Level 1', "contains(det, 'L1') and not con
addRow('FPD', "contains(det, 'FPD')")
addRow('Trigger', "contains(det, 'TRG') or contains(det, 'TFW')")
addRow('STT', "contains(det, 'STT')")
addRow('Alarm Watcher', "contains(mtype, 'alarm') and (priority > 100)")
```

An alarm display configuration file contains the filter for each row.

Identical to the alarm watcher filter.

Detector groups can generate a customized alarm display by creating a detector specific configuration file.


Left click a button to see the names of all alarms in that category.
Alarm Display
File
View
Settings
MAJOR
MINOR
INVALID
ACKED
DISABLED
GOOD
CAL and ICD
CFT
L1CTT
MUO
CFT:Minor Alarms
CFTA_AFE_4B2/VLPUD6
CFTA_AFE_4B3/AFET
CFTA_AFE_4B4/VLPUD4
CFTA_AFE_4B4/VLPUD7
CFTA_AFE_4B4/VLPUD8
CFTA_AFE_8A2/VLPUD1
CFTA_AFE_8A2/VLPUD2
CFTA_AFE_8A2/VLPUD5
CFTA_AFE_8A2/VLPUD6
CFTA_AFE_8B2/AFET
CFTA_AFE_8B6/VLPUD2
CFTA_AFE_8B6/VLPUD3
CFTC_AFE_AA7/AFET
CFTF_AFE_1A2/AFET
CFTF_AFE_CA3/AFET
CFTF_AFE_DA7/AFET
SHOW
GUIDANCE
CONTROL
ACK
ACK ALL
DISABLE
DISABLE ALL
CLOSE
MUO:Invalid Alarms
MUO_MDT_NAWB/CH03:W
MUO_PDT_011/MODE
MUO_SRC_CES/CSR0:W
SHOW
GUIDANCE
CONTROL
ACK
ACK ALL
DISABLE
DISABLE ALL
CLOSE



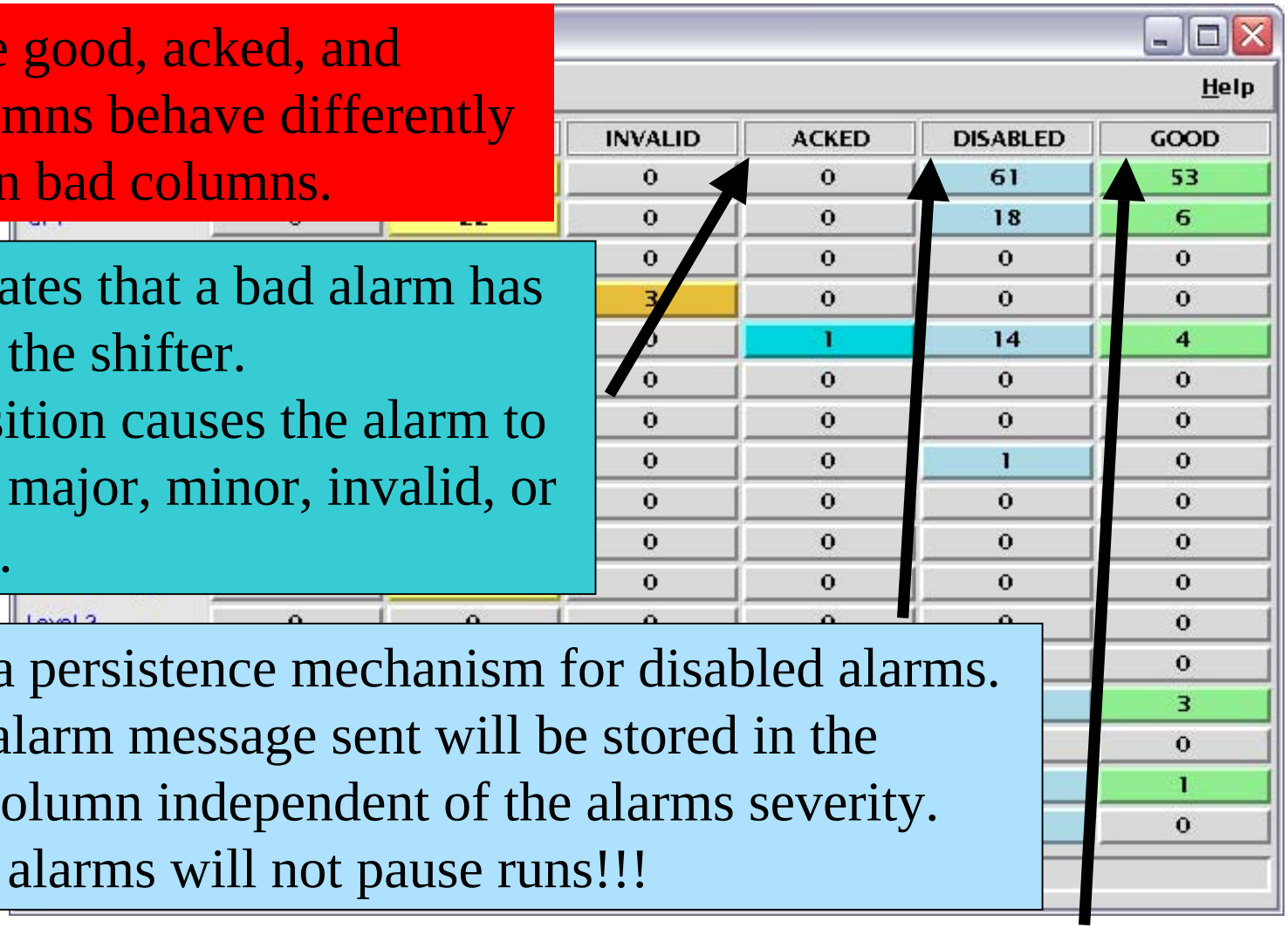
Left click a name then click the show button or double click a name to see the single alarm display.

Get more information on the alarm

Alarms can be disabled or acknowledged from here.

- Alarm names identify the source of the alarm and must be unique.
- Alarms must follow the official DZero naming convention: <detector>_<device type>_<location>/<attribute>

Alarms in the good, acked, and disabled columns behave differently than alarms in bad columns.

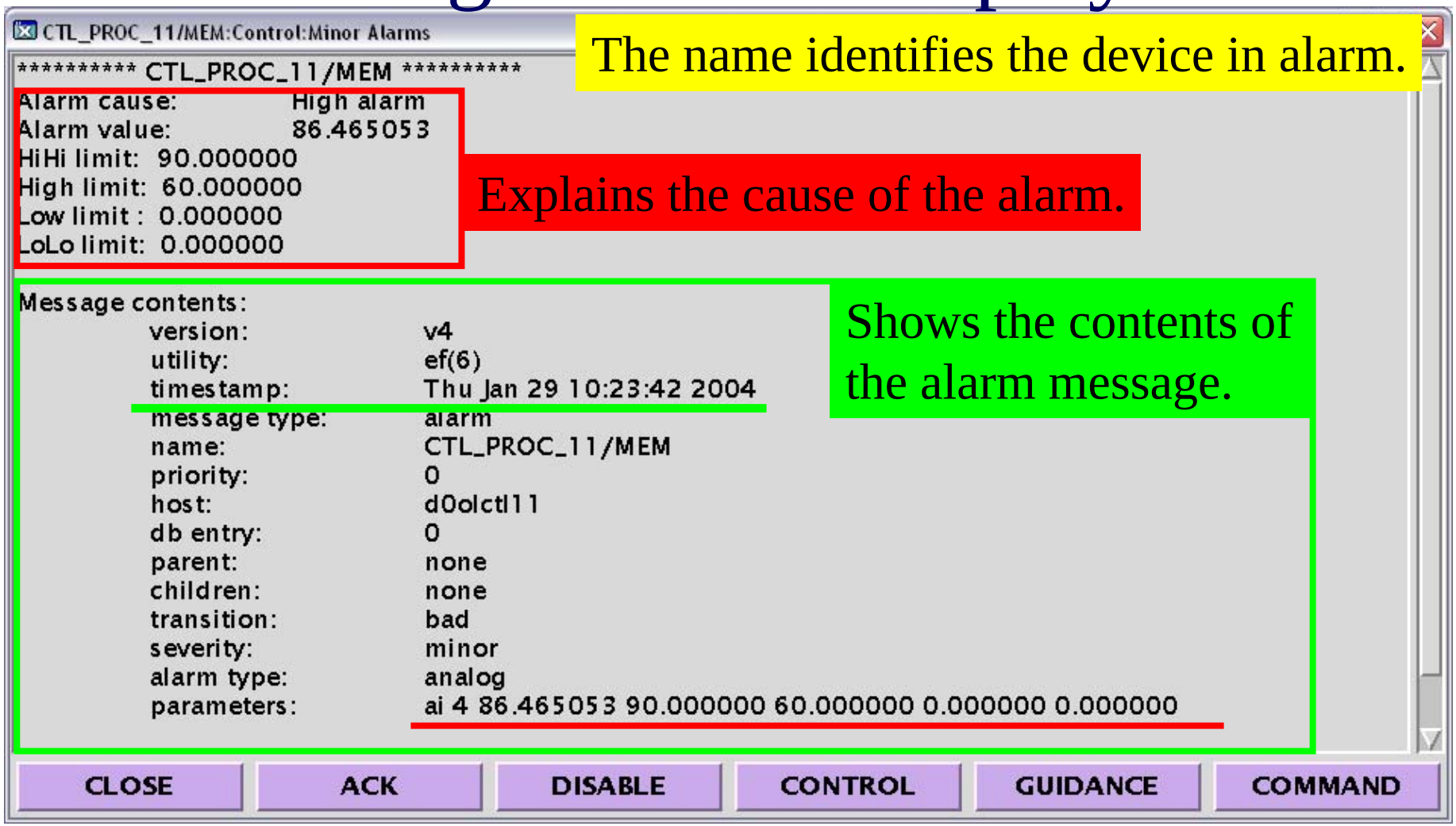


- Acked indicates that a bad alarm has been seen by the shifter.
- A state transition causes the alarm to appear in the major, minor, invalid, or good column.

- There is a persistence mechanism for disabled alarms.
- The last alarm message sent will be stored in the disabled column independent of the alarms severity.
- Disabled alarms will not pause runs!!!

- An alarm transition from bad to good is stored for five minutes.
- Multiple alarms can be listed under one name.

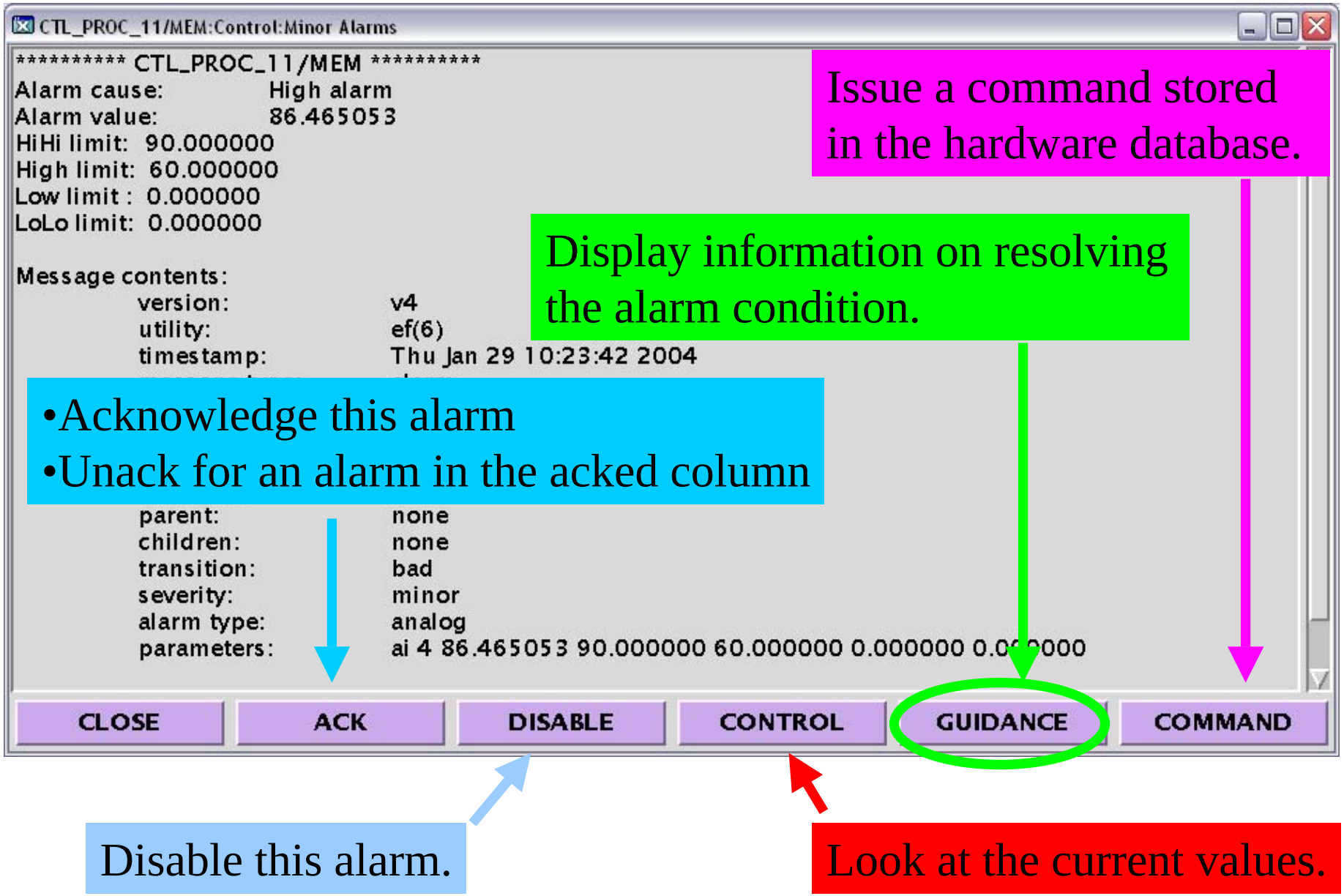
Single Alarm Display



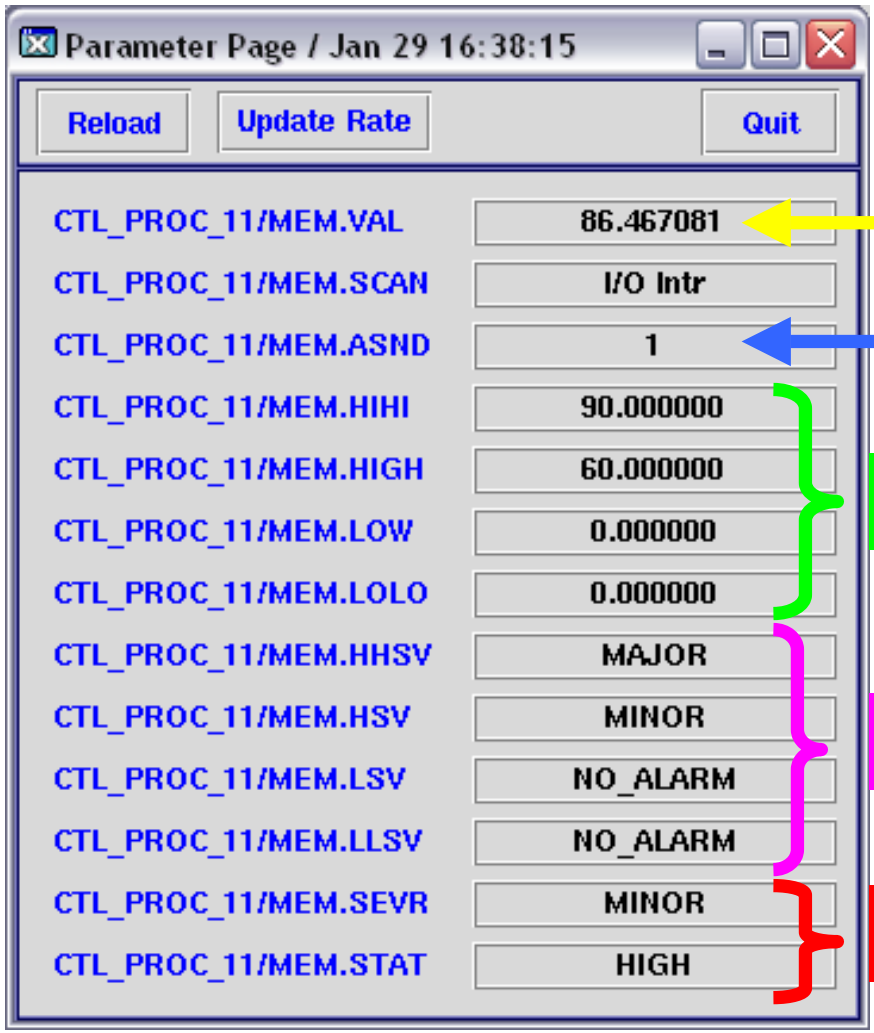
The name identifies the device in alarm.

Explains the cause of the alarm.

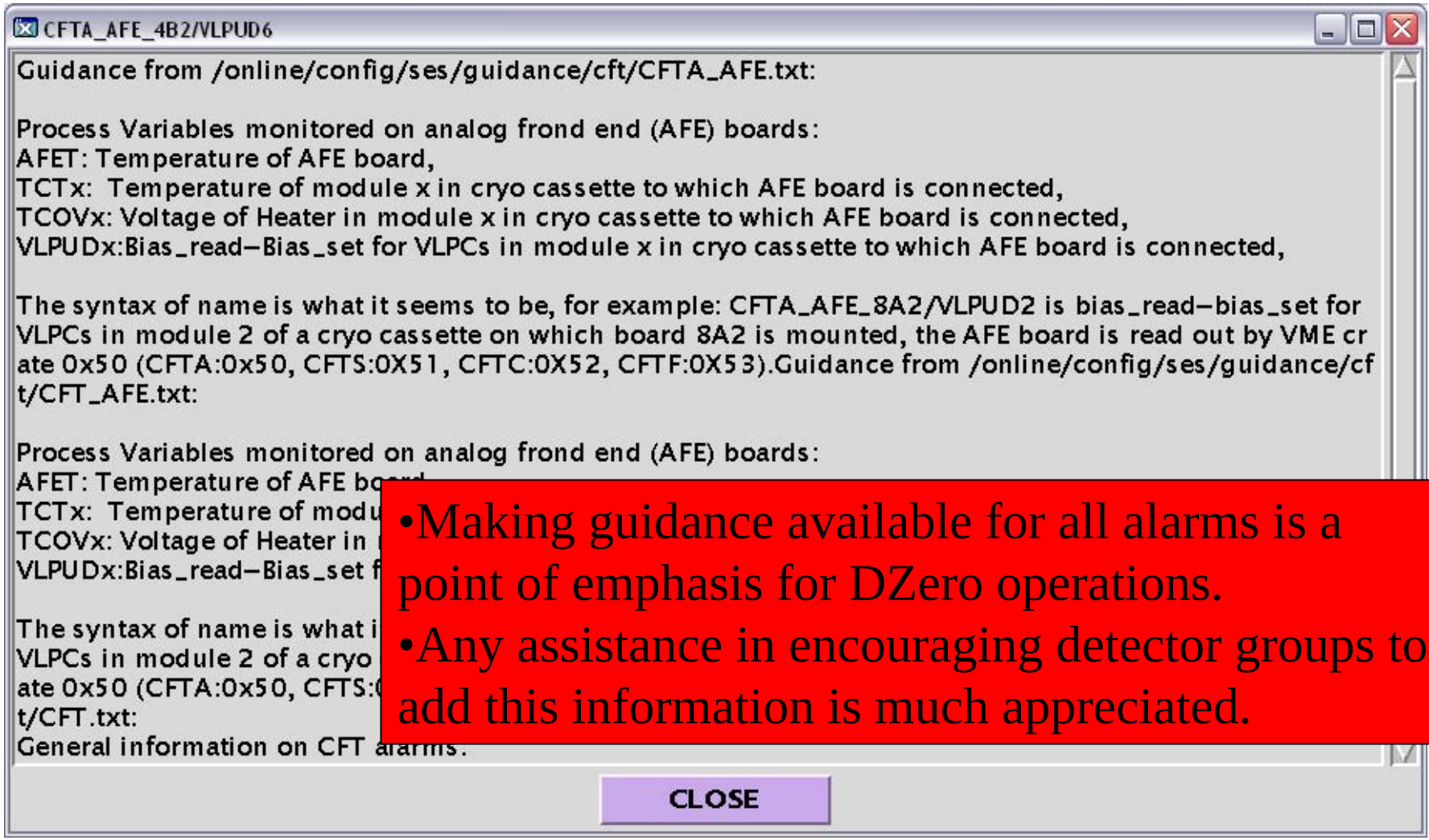
Shows the contents of the alarm message.


CTL_PROC_11/MEM:Control:Minor Alarms
********** CTL_PROC_11/MEM **********
Alarm cause: High alarm
Alarm value: 86.465053
HiHi limit: 90.000000
High limit: 60.000000
Low limit : 0.000000
LoLo limit: 0.000000
Message contents:
version: v4
utility: ef(6)
timestamp: Thu Jan 29 10:23:42 2004
parent: none
children: none
transition: bad
severity: minor
alarm type: analog
parameters: ai 4 86.465053 90.000000 60.000000 0.000000
CLOSE
ACK
DISABLE
CONTROL
GUIDANCE
COMMAND
Issue a command stored in the hardware database.
Display information on resolving the alarm condition.
•Acknowledge this alarm
•Unack for an alarm in the acked column
Disable this alarm.
Look at the current values.

Current Control Values

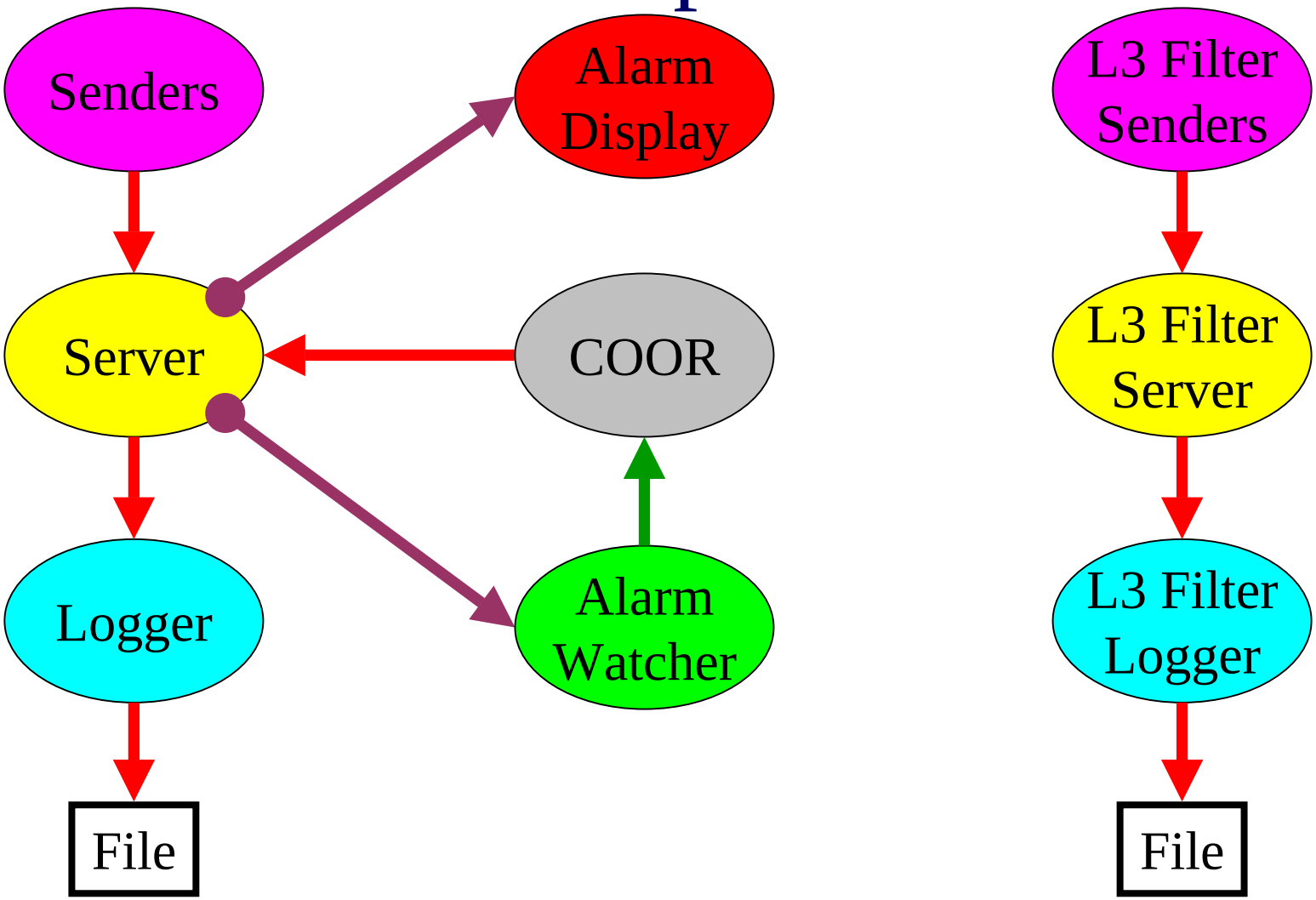


Guidance



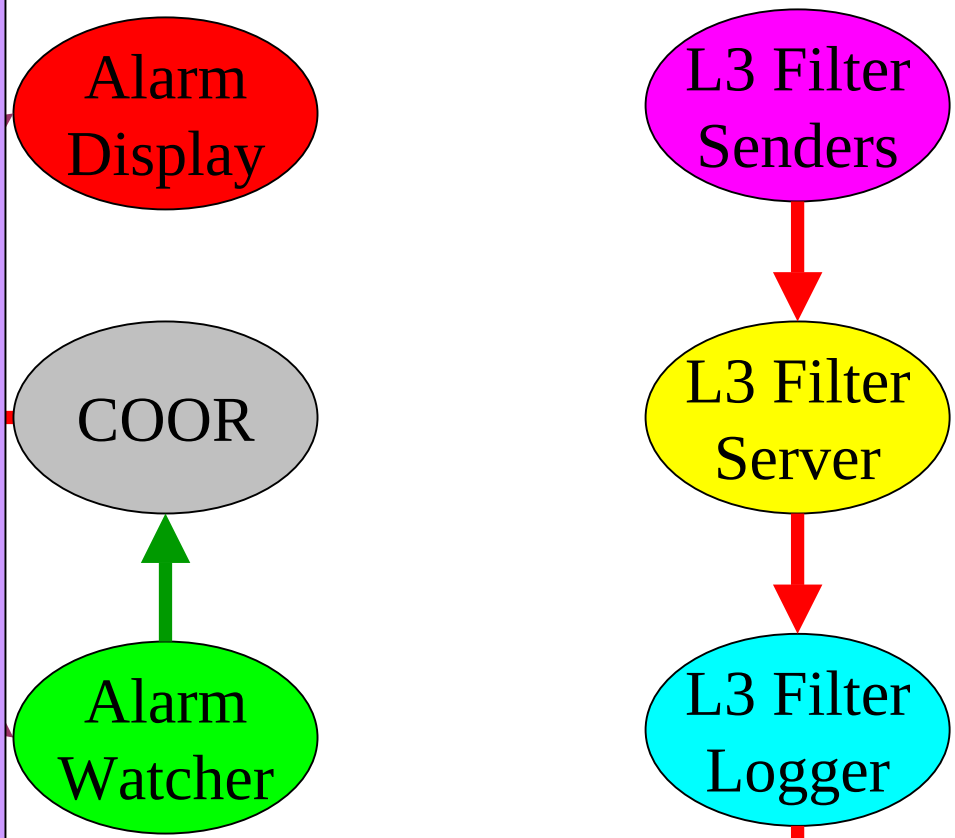
- Making guidance available for all alarms is a point of emphasis for DZero operations.
- Any assistance in encouraging detector groups to add this information is much appreciated.

Online Operations

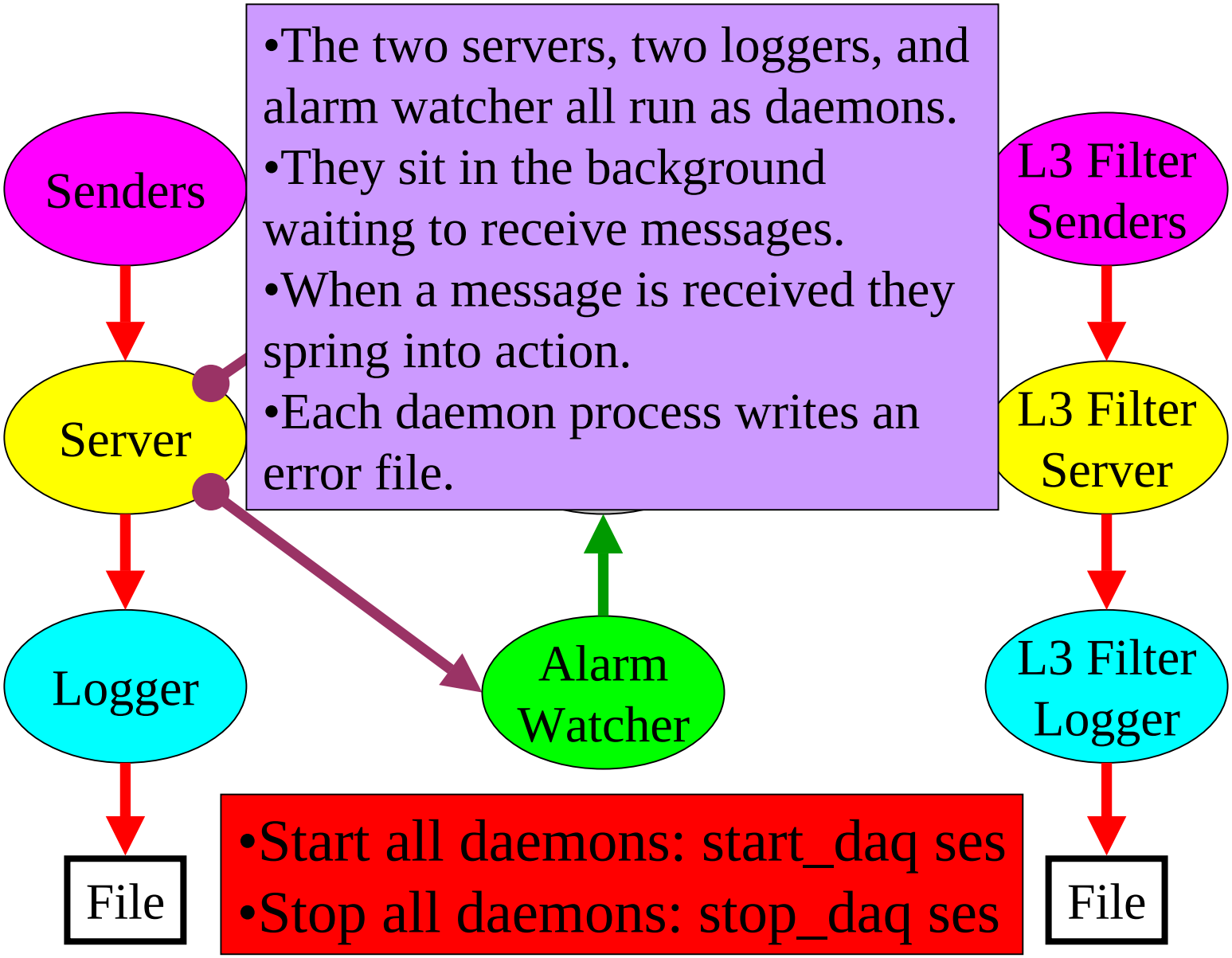


Why are there two servers in the SES?

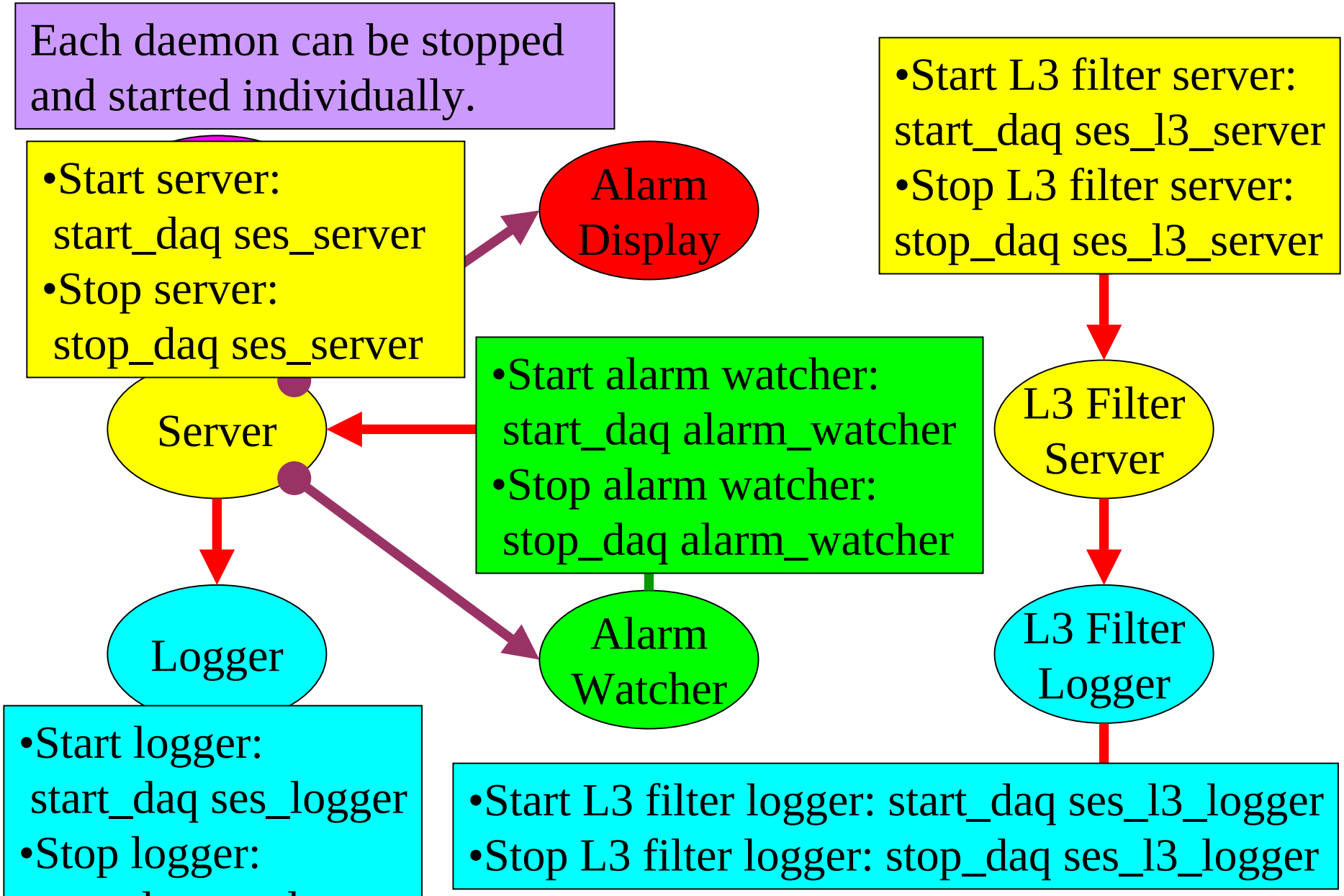
- Errors in level 3 filters are fixed without outside intervention.
- The SES framework was a simple way to get level 3 filter errors written to a file for later review.
- There is a special level 3 message that is simply passed through the server to attached clients.



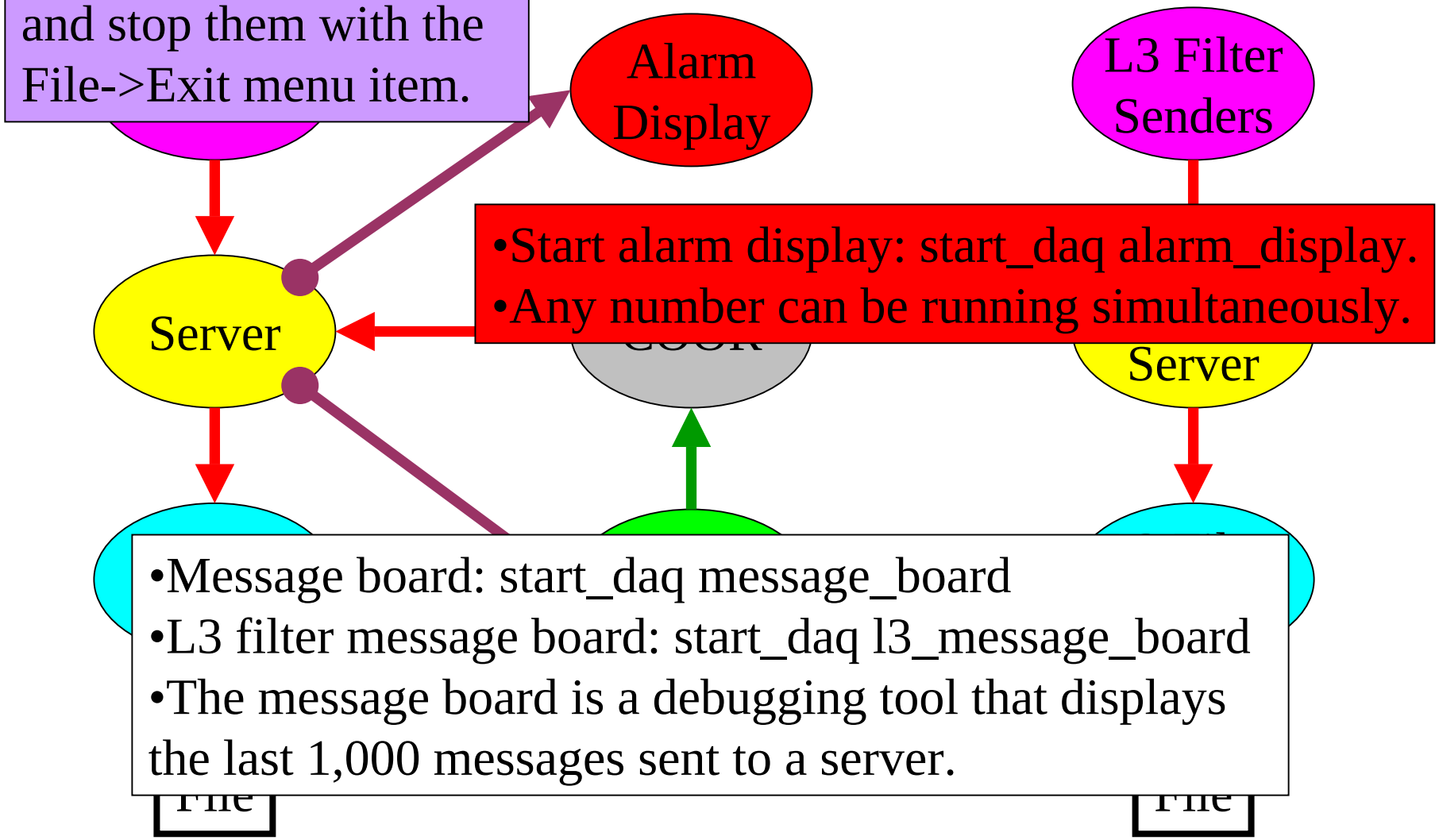
The alarm display and watcher serve no purpose connected to the level 3 filter server.


•The two servers, two loggers, and alarm watcher all run as daemons.
•They sit in the background waiting to receive messages.
•When a message is received they spring into action.
•Each daemon process writes an error file.
Senders
Server
Logger
File
Alarm Watcher
L3 Filter Senders
L3 Filter Server
L3 Filter Logger
File
•Start all daemons: start_daq ses
•Stop all daemons: stop_daq ses

Each daemon can be stopped and started individually.



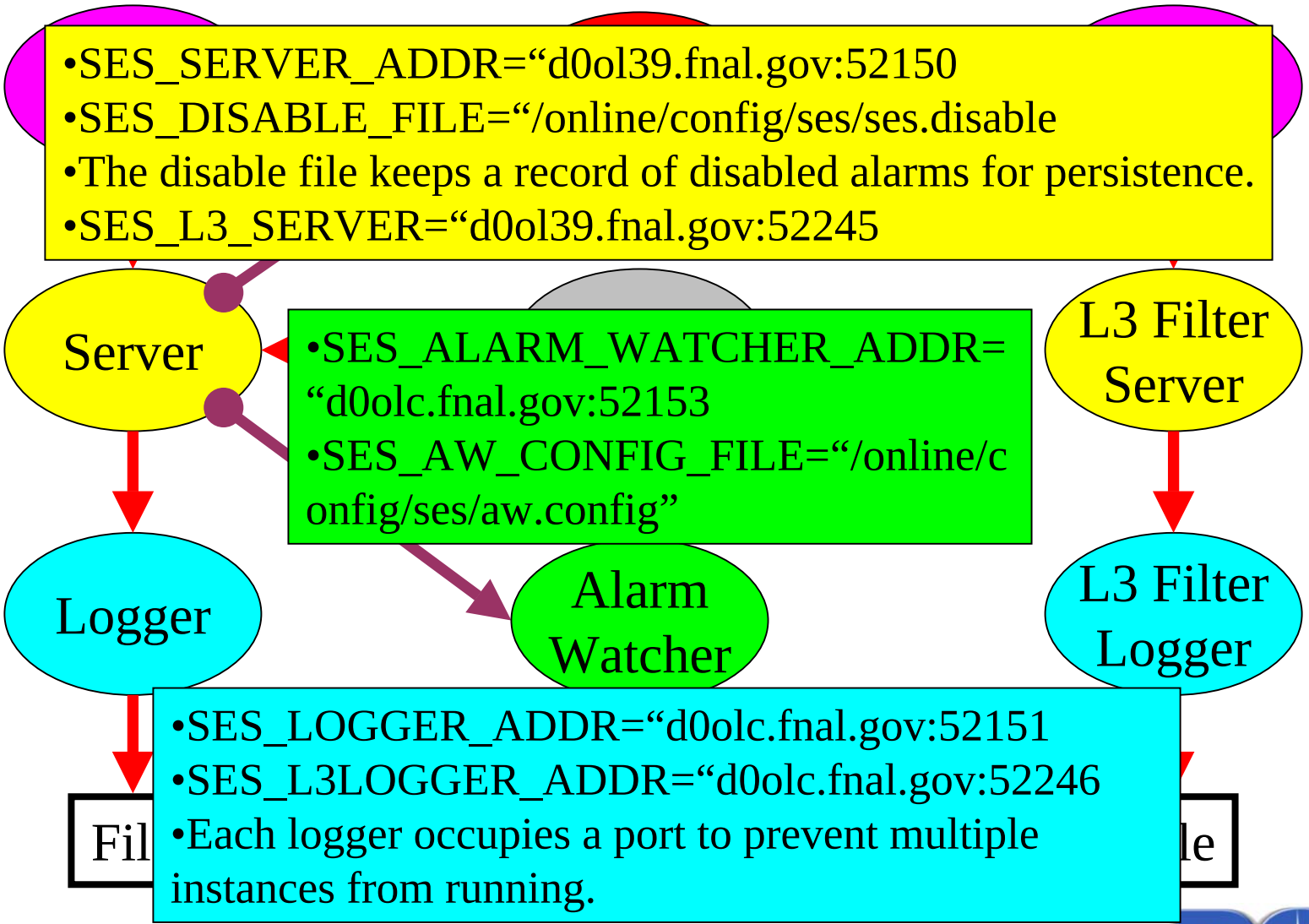
Start the user interfaces with the start_daq script and stop them with the File->Exit menu item.



- Start alarm display: start_daq alarm_display.
- Any number can be running simultaneously.

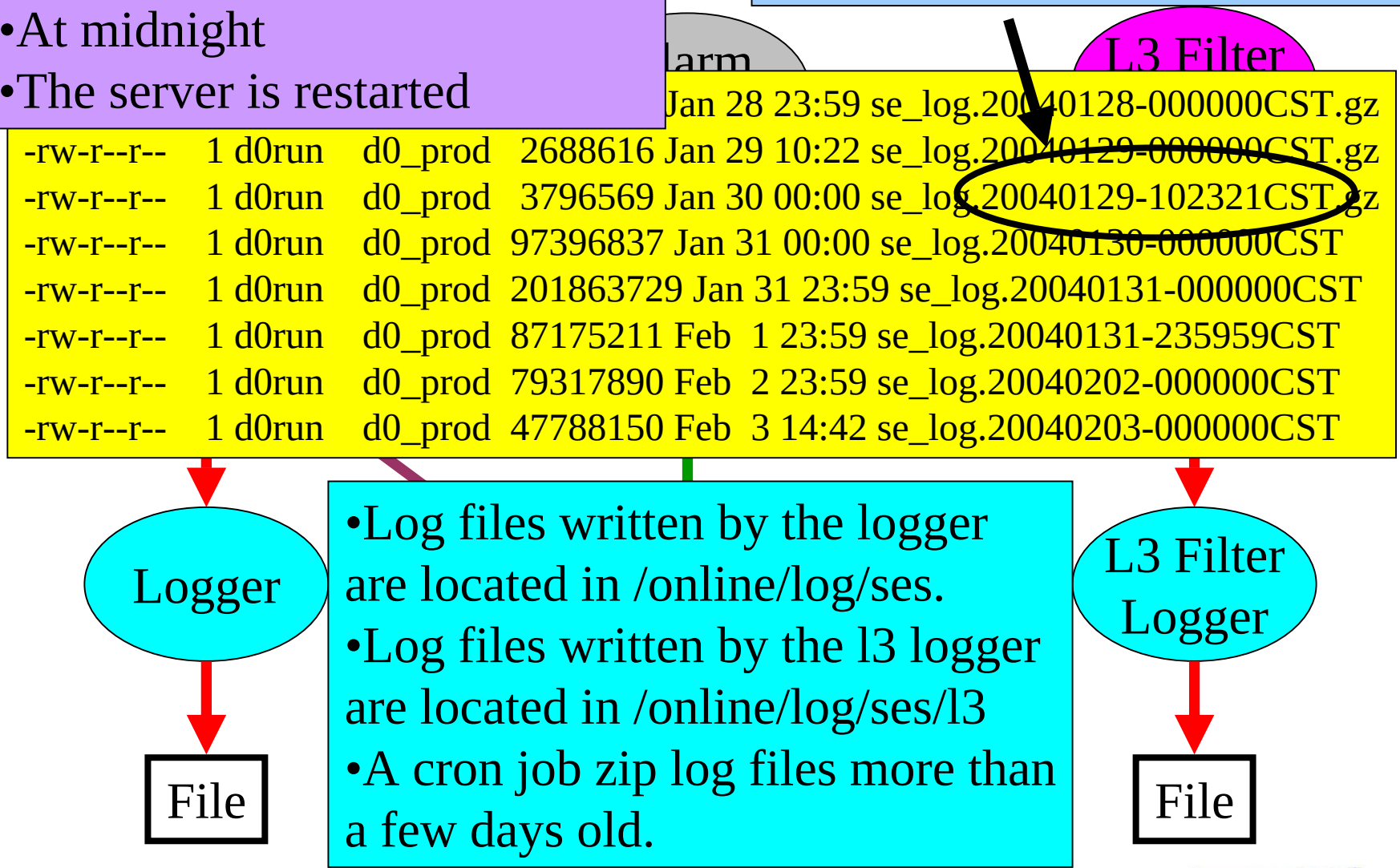
- Message board: start_daq message_board
- L3 filter message board: start_daq l3_message_board
- The message board is a debugging tool that displays the last 1,000 messages sent to a server.

System configuration is handled in /online/data/d0online/d0online_names.py.



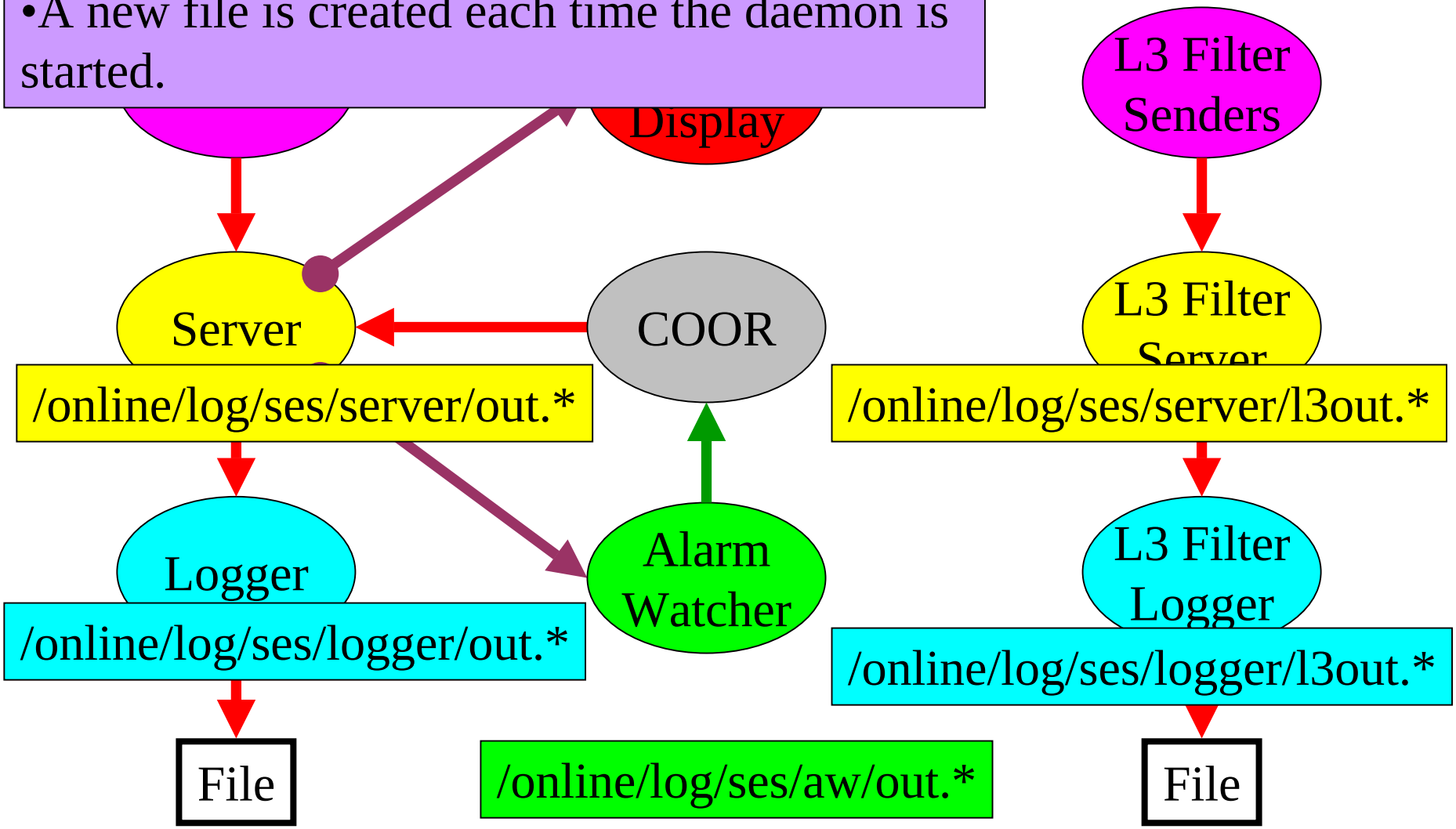
New log files are opened when:
- The logger starts
- At midnight
- The server is restarted

- Date when the file was opened.
- YYYYMMDD-NNMMSSSTZ



- Log files written by the logger are located in /online/log/ses.
- Log files written by the l3 logger are located in /online/log/ses/l3
- A cron job zip log files more than a few days old.

•All the daemons write information about their internal state to files to aid in debugging
•A new file is created each time the daemon is started.



Manually Checking the Daemons

- Log into the computer specified in d0online_names.py
- View the process status with the ps command
- When starting each daemon all the data is passed via command line arguments
  - -r = port on which a server listens for connection requests
  - -c = configuration file
  - -h = server host
  - -p = server port
  - -l = log directory
  - --coor = COOR host and port

```
<d0ol39> ps auxwww | grep seserver
d0run    29841  0.0  3.6 849036 18720 ?      S    Jan29   0:00 python
/online/products/SigEvtSys/onl04-05-00/NULL/py/seserver.py -r 52150 -c
/online/config/ses/ses.disable
d0run    31997  0.0  7.8 1036712 40084 ?     S    Jan23   0:00 python
/online/products/SigEvtSys/onl04-05-00/NULL/py/seserver.py -r 52245
```

Server

L3 Server

```
<d0olc> ps auxwww | grep selogger
d0run    612223  0.9  0.2 15.6M 6.6M ??       S      Jan 23    05:08:08 python
/online/products/SigEvtSys/onl04-05-00/NULL/py/selogger.py -p 52150 -h
d0ol39.fnal.gov -l /online/log/ses -r 52151
d0run    616384  0.2  0.2 16.3M 7.5M ??       S      Jan 23    05:57:13 python
/online/products/SigEvtSys/onl04-05-00/NULL/py/selogger.py -p 52245 -h
d0ol39.fnal.gov -l /online/log/ses/l3 -r 52246
```

Logger

L3 Logger

```
<d0olc> ps auxwww | grep watcher
d0run    687585  0.0  0.1 14.2M 5.4M ??       S      Jan 29    21:02.30 python
/online/products/SigEvtSys/onl04-05-00/NULL/py/sealarmwatcher.py -p 52150 -h
d0ol39.fnal.gov -r 52153 -c /online/config/ses/aw.config --
coor=d0olc.fnal.gov:52127 --online
```

Alarm Watcher